CLAIMS

WHAT IS CLAIMED IS:

1. An apparatus for extracting messages from a digital data stream containing messages, comprising:

a message processor that receives the digital data stream and extracts message portions from the digital data stream;

a first buffer having a plurality of locations associated with a plurality of channels to store the extracted message portions; and

a second buffer having a plurality of locations associated with the plurality of channels for storing state data corresponding to the extracted message portions.

2. The apparatus of claim 1, further comprising a central processing unit interface for coupling the apparatus to a central processing unit.

3. A device for extracting messages from a data stream, comprising:

an input interface that receives packet data in the data stream;

a packet identifier filter coupled to the input interface to selectively filter the packet data, the packet identifier filter having a central processing unit (CPU) interface to allow communication between the device and a CPU;

a message processor that receives the selectively filtered packet data from the packet identifier filter and extracts message portions from the packet data;

a first buffer having a plurality of locations associated with a plurality of channels to store the extracted message portions; and

a second buffer having a plurality of locations associated with the plurality of channels for storing state data corresponding to the extracted message portions.

4. The device of claim 3, wherein the input interface converts the packet data into parallel packet data.

5. The device of claim 4, wherein the parallel packet data is sent to the packet identifier filter with a enable signal to validate byte data in the packet.

6. The device of claim 5, wherein the input interface generates at least one clock enable signal to resynchronize the byte data.

7. The device of claim 3, wherein the packet identifier filter provides at least one selected from the group consisting of mode control, filtering control, enable control and masking control for each channel in the message processor.

8. The device of claim 7, wherein the mode control includes selecting one of a plurality of storage modes, each storage mode corresponding to a buffer size for the first buffer.

9. The device of claim 7, wherein the mode control includes selecting one of a capture mode, where the packet data is stored in the first buffer as a full packet without a sync byte, and a message mode, where messages in the packet data are allowed to be processed.

10. The device of claim 7, wherein the filtering control includes selecting whether address filtering is turned on or off, and wherein all messages in the packet data are processed when the address filtering is turned on and selected messages in the packet data are processed when the address filtering is turned off.

11. The device of claim 3, wherein the filter module has a pipeline delay to allow the packet identifier of an incoming packet to be compared with at least one predetermined packet identifier.

12. The device of claim 3, wherein the filter module validates the incoming packet by checking a header in the incoming packet with at least one predetermined condition.

13. The device of claim 3, wherein the message processor conducts a first process to find a start of a new message in the packet data and a second process to extract and store the message.

14. The device of claim 3, wherein the first buffers are circular buffers.

15. The device of claim 14, wherein the first buffer includes 32 available channels each with a 2K buffer for message storage.

16. The device of claim 14, wherein the first buffer includes 16 channels with a 2K buffer and 4 channels with an 8K buffer for message storage.

17. The device of claim 3, wherein the message processor includes:

a processor state machine shared between the plurality of channels, wherein the state data from the processor state machine is stored in the second buffer;

an address filter control circuit; and

a verification circuit that calculates a verification code and compares the calculated verification code with an embedded verification code in the message portion in the packet data.

18. The device of claim 17, wherein the message processor further includes an alternative packet capture control that stops message processing for a single channel and captures a single packet for storage in the first buffer.

19. The device of claim 17, further comprising a buffer control that controls CPU operation while the at least one of the first and second buffers is being read.

20. The device of claim 19, further comprising a message ready interrupt control coupled to the buffer control, wherein the message ready interrupt control generates signals for determining which channels have messages that are ready for processing when the CPU is interrupted based on state data in the second buffer.

21. The device of claim 17, further comprising a message error interface for identifying the presence of lost messages.

22. The device of claim 21, wherein the message error interface includes a first error circuit that identifies messages lost due to corrupt packets and a second error circuit that identifies messages lost due to first buffer overflow.

23. The device of claim 22, wherein the first and second error circuits are provided for each one of said plurality of channels.

24. A method for extracting messages from a data stream, comprising:

receiving packet data in the data stream;

selectively filtering the packet data;

extracting at least a portion of a message from the packet data;

storing said at least a portion of the message in a first buffer associated with said message processor; and

storing state data corresponding with said at least a portion of the message in a second buffer.

25. The method of claim 24, further comprising the step of converting the packet data into parallel packet data.

26. The method of claim 24, further comprising the step of providing at least one selected from the group consisting of mode control, filtering control, enable control and masking control for each channel in the message processor.

27. The method of claim 26, wherein the mode control step includes selecting one of a plurality of storage modes, each storage mode corresponding to a buffer size.

28. The method of claim 26, wherein the mode control includes selecting one of a capture mode, where the packet data is stored in the first buffer as a full packet without a sync byte, and a message mode, where messages in the packet data are allowed to be processed.

29. The method of claim 26, wherein the filtering control step includes selecting whether address filtering is turned on or off, and wherein the method includes the steps of processing all messages in the packet data when the address filtering is turned on and processing selected messages in the packet data when the address filtering is turned off.

30. The method of claim 24, further comprising the step of delaying the data stream to allow the packet identifier of an incoming packet to be compared with at least one programmed packet identifier.

31. The method of claim 24, further comprising the step of validating the incoming packet by checking a header in the incoming packet with at least one predetermined condition.

32. The method of claim 24, further comprising the steps of:

calculating a verification code; and

comparing the calculated verification code with an embedded verification code in the message in the packet data.

33. The method of claim 32, further including the steps of:

selectively stopping message processing for a single channel; and

capturing a single packet for storage in the first buffer.

34. The method of claim 33, further comprising the step of generating at least one signal for determining which channels have messages that are ready for processing when the CPU is interrupted.

35. The method of claim 32, further comprising the step of identifying the presence of lost messages due to at least one of corrupt packets and buffer overflow.